CLAIMS

WHAT IS CLAIMED IS:

1. A method for ordering equitable access to a limited resource by a plurality of contenders where each of the contenders contends for access more than one time, the method comprising:

classifying one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource;

the abused contenders attempting among themselves to gain access to the limited resource; and

repeating the above until all of the abused contenders have gained access to the limited resource.

2. The method of claim 1, wherein classifying the one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource comprises classifying the one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least a predetermined number of attempts to gain access to the limited resource.

3. The method of claim 2, further comprising:

at least a subset of the plurality of contenders attempting among themselves to gain access to the limited resource; and

determining that the one or more contenders of the plurality of contenders have failed to gain access to the limited resource at least the predetermined number of attempts.

4. The method of claim 3, further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm; and

wherein determining that the one or more contenders of the plurality of contenders have failed to gain access to the limited resource at least the predetermined number of attempts comprises determining that the one or more contenders of the plurality of contenders have failed to gain access to the limited resource at least equal to the dynamic threshold value; and

wherein classifying one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least a predetermined number of attempts to gain access to the limited resource further comprises classifying the one or more contenders of the plurality of contenders that have failed to gain access to the limited resource by at least the dynamic threshold number of attempts as abused contenders.

5. The method of claim 3, wherein at least a subset of the plurality of contenders attempting among themselves to gain access to the limited resource comprises at least the subset of the plurality of contenders attempting among themselves to gain access to the limited resource using a joust with a winner of a joust gaining access to the limited resource; and wherein determining that the one or more contenders of the plurality of contenders have failed to gain access to the limited resource at least the predetermined number of attempts comprises determining that at least a predetermined number of jousts have been lost by the one or more contenders of the plurality of contenders.

6. The method of claim 5, further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm; and

wherein determining that at least a predetermined number of jousts have been lost by the one or more contenders of the plurality of contenders further comprises determining that the dynamic threshold value has been met or exceeded by the predetermined number of jousts that have been lost by the one or more of the plurality of contenders; and

wherein classifying one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least a predetermined number of attempts to enter the list further comprises classifying the one or more contenders of the plurality of contenders that have failed to gain access to the limited resource by at least the dynamic threshold number of attempts as abused contenders.

7. The method of claim 2, further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm.

8. The method of claim 1, further comprising:

starting a sequence when at least one of the abused contenders is first identified;

ending the sequence when all abused contenders have gained access to the limited resource.

9. The method of claim 8, further comprising:

maintaining a history of the abused contenders that have gained access to the limited resource during the sequence; and

resetting the history of the abused contenders at the ending of the sequence.

10. The method of claim 9, further comprising:

maintaining a record of the contenders that become newly abused contenders after the sequence has already started; and

starting a new sequence immediately after the ending of the sequence with the newly abused contenders as the abused contenders for the new sequence.

11. The method of claim 10, further comprising:

maintaining a history of the abused contenders that have gained access to the limited resource during the new sequence; and

resetting the history of the abused contenders at the ending of the new sequence.

12. A method for acquiring a spinlock in a computer system, the method comprising:

classifying one or more of a plurality of devices that have failed to acquire the spinlock as abused devices;

the abused devices attempting among themselves repeatedly to acquire the spinlock; and

repeating the above until all of the abused devices have acquired the spinlock.

13. The method of claim 12, wherein classifying one or more of the plurality of devices that have failed to acquire the spinlock as the abused devices comprises classifying the one or more of the plurality of devices that have failed to acquire the spinlock for at least a predetermined number of attempts as the abused devices

14. The method of claim 13, further comprising:

at least a subset of the plurality of devices attempting among themselves to acquire the spinlock; and

determining that the one or more of the plurality of devices have failed to acquire the spinlock at least the predetermined number of attempts.

15. The method of claim 14, further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm; and

wherein determining that the one or more of the plurality of devices have failed to acquire the spinlock at least the predetermined number of attempts comprises determining that the one or more of the plurality of devices have failed to acquire the spinlock by a number of attempts that equals or exceeds the dynamic threshold value; and

wherein classifying one or more of a plurality of devices that have failed to acquire the spinlock for at least a predetermined number of attempts as abused devices comprises classifying the one or more of the plurality of devices that have failed to acquire the spinlock by at least the dynamic threshold number of attempts as abused devices.

16. The method of claim 14, wherein at least a subset of the plurality of devices attempting among themselves to acquire the spinlock further comprises at least a subset of the plurality of devices attempting among themselves to acquire the spinlock using a joust with a winner of the joust acquiring the spinlock; and

wherein determining that the one or more of the plurality of devices have failed to acquire the spinlock at least the predetermined number of attempts further comprises determining

that at least a predetermined number of jousts have been lost by the one or more of the plurality of devices.

17. The method of claim 16, further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm; and

wherein determining that at least a predetermined number of jousts have been lost by the one or more of the plurality of devices further comprises determining that the one or more of the plurality of devices have lost the joust by a number of attempts that equals or exceeds the dynamic threshold value; and

wherein classifying one or more of a plurality of devices that have failed to acquire the spinlock for at least a predetermined number of attempts as abused devices comprises classifying the one or more of the plurality of devices that have failed to acquire the spinlock by at least the dynamic threshold number of attempts as abused devices.

18. The method of claim 13, further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm.

19. The method of claim 12, further comprising:

starting a sequence when any abused device is first identified;

ending the sequence when all abused devices have acquired the spinlock.

20. The method of claim 19, wherein ending the sequence when all abused devices have acquired the spinlock comprises ending the sequence when all abused devices have acquired the spinlock only once.

21. The method of claim 19, further comprising:

maintaining a history of the abused devices that have acquired the spinlock during the sequence; and

resetting the history of the abused devices that have acquired the spinlock at the ending of the sequence.

22. The method of claim 19, further comprising:

maintaining a record of the devices that become newly abused devices after the sequence has already started; and

starting a new sequence immediately after the ending of the sequence with the newly abused devices as the abused devices for the new sequence.

23. The method of claim 22, further comprising:

maintaining a history of the abused devices that have acquired the spinlock during the new sequence; and

resetting the history of the abused devices at the ending of the new sequence.

24. A data structure associated with a spinlock, the data structure comprising:

an abuse bitmask comprising a first plurality of data entries, one for each processor, which indicate whether a given processor is an abused processor;

a history bitmask comprising a second plurality of data entries, one for each processor, which indicate whether the abused processor has acquired the spinlock since becoming abused; and

an abuse threshold entry that indicates how many times a given processor must attempt to acquire the spinlock and fail to acquire the spinlock before becoming abused.

25. The data structure of claim 24, further comprising:

a release count entry that indicates how many times the spinlock has been acquired and released.

26. The data structure of claim 24, wherein the threshold value is a static threshold value and does not change for a period of time.

27. The data structure of claim 24, wherein the threshold value is a dynamic threshold value and does change based on a predetermined algorithm.

28. The data structure of claim 27, wherein the dynamic threshold value depends on how many processors are in the computer system, and wherein the dynamic threshold value is modified upon a processor entering or leaving the computer system including the data structure.

29. The data structure of claim 24, further comprising:

one or more flag entries for storing values corresponding to one or more characteristics associated with the data structure.

30. The data structure of claim 29, wherein the one or more flag entries comprise a bit to indicate if the threshold value is static or dynamic.

31. The data structure of claim 29, wherein the one or more flag entries comprise one or more bits to indicate performance or debugging values.

32. The data structure of claim 24, wherein the data structure is stored in a cache line in a cache memory.

33. An array of data structures, the array of data structures comprising a plurality of the data structures according to claim 24, wherein each of the plurality of the data structures according to claim 24 is associated either with a different spinlock or with one or more spinlocks of a same rank.

34. A computer readable program storage device encoded with instructions that, when executed by a computer, performs a method for ordering equitable access to a limited resource by a plurality of contenders where each of the contenders contends for access more than one time, the method comprising:

tracking a number of unsuccessful attempts by one or more of the plurality of contenders to gain access to the limited resource;

comparing the number of unsuccessful attempts by the one or more of the plurality of contenders to access the limited resource to a threshold value;

classifying the one or more of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least the threshold value number of attempts to gain access to the limited resource; and

limiting contention for access to the limited resource to the abused contenders.

35. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 34, the method further comprising:

repeating the above until all of the abused contenders have gained access to the limited resource.

36. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 35, the method further comprising:

tracking a number of attempts by at least a subset of the plurality of contenders to gain access to the limited resource.

37. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 36, the method further comprising:

dynamically modifying the threshold value to a dynamic threshold value using a predetermined algorithm;

wherein comparing the number of unsuccessful attempts by the one or more of the plurality of contenders to access the limited resource to a threshold value comprises comparing the number of unsuccessful attempts by the one or more of the plurality of contenders to access the limited resource to the dynamic threshold value; and

wherein classifying the one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least the

threshold value number of attempts to gain access to the limited resource further comprises classifying the one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least the dynamic threshold value number of attempts to gain access to the limited resource.

38. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 36,

wherein tracking a number of attempts by at least a subset of the plurality of contenders to gain access to the limited resource further comprises tracking a number of attempts by at least a subset of the plurality of contenders to gain access to the limited resource using a joust with a winner of the joust gaining access to the limited resource; and

wherein tracking a number of unsuccessful attempts by one or more of the plurality of contenders to gain access to the limited resource further comprises tracking a number of unsuccessful jousts by the one or more of the plurality of contenders to gain access to the limited resource.

39. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 38, the method further comprising:

dynamically modifying the threshold value to a dynamic threshold value using a predetermined algorithm;

wherein comparing the number of unsuccessful attempts by the one or more of the plurality of contenders to access the limited resource to a threshold value further comprises comparing the number of unsuccessful jousts by the one or more of the plurality of contenders to access the limited resource to the dynamic threshold value; and

wherein classifying the one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least the threshold value number of attempts to gain access to the limited resource further comprises classifying the one or more contenders of the plurality of contenders as abused contenders that have lost the joust at least the dynamic threshold value number of attempts.

40. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 34, the method further comprising:

starting a sequence when at least one of the abused contenders is first identified;

ending the sequence when all abused contenders have gained access to the limited resource.

41. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 40, the method further comprising:

maintaining a history of the abused contenders that have gained access to the limited resource during the sequence; and

resetting the history of the abused contenders at the ending of the sequence.

42. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 41, the method further comprising:

maintaining a record of the contenders that become newly abused contenders after the sequence has already started; and

starting a new sequence immediately after the ending of the sequence with the newly abused contenders as the abused contenders for the new sequence.

43. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 42, the method further comprising:

maintaining a history of the abused contenders that have gained access to the limited resource during the new sequence; and

resetting the history of the abused contenders at the ending of the new sequence.

44. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 34, the method further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm.

45. A computer readable program storage device encoded with instructions that, when executed by a computer, performs a method for acquiring a spinlock in a computer system where each device contends for the spinlock more than one time, the method comprising:

tracking a number of unsuccessful attempts by one or more of a plurality of devices to acquire the spinlock;

comparing the number of unsuccessful attempts by the one or more of the plurality of devices to access the spinlock to a threshold value;

classifying the one or more of the plurality of devices as abused devices that have failed to acquire the spinlock after at least the threshold value number of attempts to acquire the spinlock; and

limiting contention for the spinlock to the abused devices.

46. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 45, the method further comprising:

repeating the above until all of the abused devices have acquired to the spinlock.

47. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 46, the method further comprising:

tracking a number of attempts by at least a subset of the plurality of devices to acquire the spinlock.

48. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 47, the method further comprising:

dynamically modifying the threshold value to a dynamic threshold value using a predetermined algorithm;

wherein comparing the number of unsuccessful attempts by the one or more of the plurality of devices to access the spinlock to a threshold value further comprises comparing the number of unsuccessful attempts by the one or more of the plurality of devices to access the spinlock to the dynamic threshold value; and

wherein classifying the one or more devices of the plurality of devices as abused devices that have failed to acquire the spinlock after at least the threshold value number of attempts to acquire the spinlock further comprises classifying the one or more devices of the plurality of devices as abused devices that have failed to acquire the spinlock after at least the dynamic threshold value number of attempts to acquire the spinlock.

49. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 47,

wherein tracking a number of attempts by at least a subset of the plurality of devices to acquire the spinlock further comprises tracking a number of attempts by at least a subset of the plurality of devices to acquire the spinlock using a joust with a winner of the joust gaining access to the spinlock; and

wherein tracking a number of unsuccessful attempts by one or more of the plurality of devices to acquire the spinlock further comprises tracking a number of unsuccessful jousts by the one or more of the plurality of devices to acquire the spinlock.

50. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 49, the method further comprising:

dynamically modifying the threshold value to a dynamic threshold value using a predetermined algorithm;

wherein comparing the number of unsuccessful attempts by the one or more of the plurality of devices to access the spinlock to a threshold value further comprises comparing the number of unsuccessful jousts by the one or more of the plurality of devices to access the spinlock to the dynamic threshold value; and

wherein classifying the one or more devices of the plurality of devices as abused devices that have failed to acquire the spinlock after at least the threshold value number of attempts to acquire the spinlock further comprises classifying the one or more devices of the plurality of devices as abused devices that have lost the joust at least the dynamic threshold value number of attempts.

51. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 45, the method further comprising:

starting a sequence when at least one of the abused devices is first identified;

ending the sequence when all abused devices have gained access to the spinlock.

52. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 51; the method further comprising:

maintaining a history of the abused devices that have gained access to the spinlock during the sequence; and

resetting the history of the abused devices at the ending of the sequence.

53. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 52, the method further comprising:

maintaining a record of the devices that become newly abused devices after the sequence has already started; and

starting a new sequence immediately after the ending of the sequence with the newly abused devices as the abused devices for the new sequence.

54. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 53, the method further comprising:

maintaining a history of the abused devices that have gained access to the spinlock during the new sequence; and

resetting the history of the abused devices at the ending of the new sequence.

55. The computer readable program storage device encoded with instructions that, when executed by the computer, performs the method as described in claim 45, the method further comprising:

dynamically modifying the predetermined number to a dynamic threshold value using a predetermined algorithm.

56. A computer system, comprising:

at least one shared resource with an associated spinlock;

a plurality of processors configured to access the at least one shared resource using the associated spinlock;

a memory; and

a data structure encoded on the memory and associated with the associated spinlock, the data structure comprising:

an abuse bitmask comprising a first plurality of data entries, one for each of the plurality of processors, which indicate whether a given processor is an abused processor;

a history bitmask comprising a second plurality of data entries, one for each of the plurality of processors, which indicate whether the abused processor has acquired the associated spinlock since becoming abused; and

an abuse threshold entry that indicates how many times a given processor must attempt to acquire the spinlock and fail to acquire the associated spinlock before becoming abused.

57. The computer system of claim 56, wherein the memory is a cache memory.

58. The computer system of claim 57, wherein the data structure is stored in a cache line of the cache memory.

59. The computer system of claim 56, wherein the plurality of processors are taken from the group consisting of: microprocessors, digital signal processors, and controllers.

60. The computer system of claim 56, wherein each of the plurality of processors is substantially identical.

61. The computer system of claim 56, wherein one or more of the plurality of processors is substantially different.

62. The computer system of claim 56, wherein the data structure further comprises:

a release count entry that indicates how many times the spinlock has been acquired and released.

63. The computer system of claim 56, wherein the threshold value is a static threshold value and does not change for a period of time.

64. The computer system of claim 56, wherein the threshold value is a dynamic threshold value and does change based on a predetermined algorithm.

65. The computer system of claim 64, wherein the dynamic threshold value depends on how many processors are in the computer system, and wherein the dynamic threshold value is modified upon a processor entering or leaving a computer system including the data structure.

66. The computer system of claim 56, wherein the data structure further comprises:

one or more flag entries for storing values corresponding to one or more characteristics associated with the data structure.

67. The computer system of claim 66, wherein the one or more flag entries comprise a bit to indicate if the threshold value is static or dynamic.

68. The computer system of claim 66, wherein the one or more flag entries comprise one or more bits to indicate performance or debugging values.

69. The computer system of claim 56, wherein the memory includes a cache memory, and wherein the data structure is stored in a cache line in the cache memory.

70. The computer system of claim 56, wherein the at least one shared resource with an associated spinlock comprises a plurality of shared resources with a corresponding plurality of associated spinlocks, a respective spinlock associated with each of the plurality of shared resources; wherein the plurality of processors are configured to access each of the plurality of shared resources, using the associated spinlock; and wherein the data structure is comprises in an array of data structures, the array of data structures comprising a plurality of the data structure, wherein each of the plurality of the data structure is associated with a different spinlock.

71. A method for ordering equitable access to a limited resource by a plurality of contenders where each of the contenders contends for access more than one time, the method comprising:

step for classifying one or more contenders of the plurality of contenders as abused contenders that have failed to gain access to the limited resource after at least a predetermined number of attempts to gain access to the limited resource;

step for the abused contenders attempting among themselves to gain access to the limited resource; and

step for repeating the above until all of the abused contenders have gained access to the limited resource.

72. A method for acquiring a spinlock in a computer system, the method comprising:

step for classifying one or more of a plurality of devices that have failed to acquire the spinlock for at least a predetermined number of attempts as abused devices;

step for the abused devices attempting among themselves repeatedly to acquire the spinlock; and

step for repeating the above until all of the abused devices have acquired the spinlock.

73. A data structure associated with a spinlock, the data structure comprising:

means for indicating whether a given processor is an abused processor;

means for indicating whether the abused processor has acquired the spinlock since becoming abused; and

means for indicating how many times a given processor must attempt to acquire the spinlock and fail to acquire the spinlock before becoming abused.

74. A computer system, comprising:

means for sharing a resource with means for processing;

means for locking the means for sharing the resource, the means for locking the means for sharing the resource being associated with the means for sharing the resource;

the means for processing configured to access the means for sharing using the means for locking to lock the means for sharing; and

means for storing data associated with the means for sharing, wherein the data comprise:

a first plurality of entries, one for each means for processing, which indicate whether a given means for processing is an abused means for processing;

a second plurality of entries, one for each means for processing, which indicate whether the abused means for processing has acquired the means for locking since becoming abused; and

an entry to indicate how many times a given means for processing must attempt to acquire the means for locking and fail to acquire the means for locking before becoming abused.

75. For use in a computer system comprising a plurality of processors, each including a data structure associated with a spinlock, the data structure comprising an abuse bitmask comprising a first plurality of data entries, one for each processor configured to access the spinlock, which indicate whether a given processor is an abused processor; a history bitmask comprising a second plurality of data entries, one for each processor configured to access the spinlock, which indicate whether the abused processor has acquired the spinlock since becoming abused; an abuse threshold entry that indicates how many times a given processor must attempt to acquire the spinlock and fail to acquire the spinlock before becoming abused,

and a release count entry that indicates how many times the spinlock has been acquired and released; a method for acquiring the spinlock, the method comprising:

setting the abuse threshold entry;

incrementing the release count entry each time the spinlock is acquired and released;

noting a current value for the release count entry for each processor of the plurality of processors that fails to acquire the spinlock;

determining that one or more of the plurality of processors that have failed to acquire the spinlock for at least the abuse threshold entry number of attempts as abused processors;

setting a respective one or more of the first plurality of data entries of the data structure corresponding to the one or more of the plurality of processors that are the abused processors;

setting a respective one of the second plurality of data entries of the data structure corresponding to the abused processor that acquires the spinlock from the abused processors attempting among themselves repeatedly to acquire the spinlock until all of the abused processors have acquired the spinlock; and

clearing the abuse bitmask and the history bitmask after all of the abused processors have acquired the spinlock.

76. The method according to claim 75, further comprising:

noting one or more processors of the plurality of processors that failed to acquire the spinlock for at least the abuse threshold entry number of attempts as newly abused processors;

setting a respective one or more of the first plurality of data entries of the data structure corresponding to the newly abused processors after clearing the abuse bitmask and the history bitmask after all of the abused processors have acquired the spinlock;

setting a respective one of the second plurality of data entries of the data structure corresponding to the newly abused processor that acquires the spinlock from the newly abused processors attempting among themselves repeatedly to acquire the spinlock until all of the newly abused processors have acquired the spinlock; and

clearing the abuse bitmask and the history bitmask after all of the newly abused processors have acquired the spinlock.